COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A RIGID STRUCTURE

(NASA-TM-78681) COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A RIGID STRUCTURE (NASA) 47 F HC A03/MF A01

CSCL 20K Unclas
G3/39 11902

REID A. HULL, JOHN L. GILBERT AND PHILLIP J. KLICH

MARCH 1978

NASA
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665
A computer program (ND0702) has been developed for the rapid computation of the mass properties of complex structural systems. The program uses rigid body analyses and permits differences in structural material throughout the total system. It is based on the premise that complex systems can be adequately described by a combination of basic elemental shapes. The following thirteen widely used structural shapes were selected for inclusion in the program:

1. Discrete mass
2. Cylinder
3. Truncated cone
4. Torus
5. Beam (arbitrary cross section)
6. Circular rod (arbitrary cross section)
7. Spherical segment
8. Sphere
9. Hemisphere
10. Parallelepiped
11. Swept Trapezoidal Panel
12. Symmetric Trapezoidal Panels
13. Curved Rectangular Panel

Simple geometric data describing size and location of each element and the respective material density or weight of each element are the only required input data. From this minimum input, the program yields system weight, center of gravity, moments of inertia and products of inertia with respect to mutually perpendicular axes through the system center of gravity. The program also yields mass properties of the individual shapes relative to component axes.

Permanent configuration records and the use of iterative calculations to investigate design systems or to determine optimums contribute to the cost-effectiveness of the program's use.

INTRODUCTION

Determining the mass properties of any rigid structure is a problem that at times becomes complex, but one which can easily be dealt with utilizing computer solutions.
For rigid structures the solution of the mass properties requires transformation to an axis parallel to the system axis and becomes laborious almost to the point of being impractical.

Any complex structure must be broken down into elements in order to exact a solution. The approach selected for the program presented in this paper was to automate the input to the point where an element's shape, geometry, density or weight, and three grid points are the only requirements. This approach was influenced by the simplicity of computing the direction cosines (Euler angle relationship) from the given three grid points. The program as outlined in this paper performs essentially the same process as calculations "by hand" and is extremely useful for rigid structures skewed in space. This program also provides improved accuracy, time savings, and complete permanent records for a mass properties analysis. (This TMX is a verified expansion of the LaRC working paper "Computer Program for Determining Mass Properties of a Composite Body", by Phillip J. Klich and John L. Gilbert dated Oct. 22, 1968.)

SYMBOLS

\( I \) \[ \text{moments and products of inertia} \]
\( \mathbf{r} \) \[ \text{displacement vector for differential mass} \]
\( T \) \[ \text{kinetic energy} \]
\( \mathbf{v} \) \[ \text{linear velocity} \]
\( x', y', z' \) \[ \text{rectangular coordinate component displacement vectors} \]
\( x, y, z \) \[ \text{rectangular coordinate system displacement vectors} \]
\( \mathbf{w} \) \[ \text{angular velocity} \]
\( 'x'x \) \[ \text{cosine of angle between } x' \text{ and } x \text{ axes} \]
\( 'y'x \) \[ \text{cosine of angle between } y' \text{ and } x \text{ axes} \]
\( 'z'x \) \[ \text{cosine of angle between } z' \text{ and } x \text{ axes} \]
\( 'x'y \) \[ \text{cosine of angle between } x' \text{ and } y \text{ axes} \]
\[ l_{y'y} \text{ cosine of angle between } y' \text{ and } y \text{ axes} \]
\[ l_{z'y} \text{ cosine of angle between } z' \text{ and } y \text{ axes} \]
\[ l_{x'z} \text{ cosine of angle between } x' \text{ and } z \text{ axes} \]
\[ l_{y'z} \text{ cosine of angle between } y' \text{ and } z \text{ axes} \]
\[ l_{z'z} \text{ cosine of angle between } z' \text{ and } z \text{ axes} \]

Subscripts

- \( co \): system coordinates to component center of mass
- \( xx \): refer to component axis to which moments of inertia are calculated
- \( yy \): refer to system axis to which the moments of inertia are rotated parallel to the system coordinates

Superscripts

- \( prime \): denotes component coordinate system

INERTIA EQUATIONS

The mass properties of shapes such as a cylinder, sphere, etc., are easily calculated and therefore were selected as the basic component shapes for handling a system such as a spacecraft structure. Since the component shape mass properties are measured with respect to their respective center of mass, these properties have to be transferred to the system center of mass. The transformation can be made in two steps: first, the component properties are transferred to a system parallel to the system axis, and then transferred by the usual parallel axis theorem. The rotational transformation is derived by using the principle of kinetic energy. An introductory derivation of the moment of inertia, and product of inertia expressions are derived first and then transformation from the component to system coordinates is presented.

Derivation of Inertia Equations

Using the expressions of kinetic energy of a rigid body, the equations of moments of inertia and products of inertia are derived. Consider a component spinning with an angular velocity \( \omega \) as shown next.
The angular kinetic energy can be written as

\[ T = \frac{1}{2} \int \mathbf{V} \cdot \mathbf{V} \, dm \]

where the velocity is expressed as

\[ \mathbf{V} = \mathbf{\omega} \times \mathbf{r} \]

Substituting in the kinetic energy expression gives

\[ T = \frac{1}{2} \int (\mathbf{\omega} \times \mathbf{r}) \cdot (\mathbf{\omega} \times \mathbf{r}) \, dm \]

where

\[ \mathbf{\omega} = i\omega_x + j\omega_y + k\omega_z \]

and

\[ \mathbf{r} = ix' + jy' + kz' \]

Taking the cross product

\[
\begin{vmatrix}
1 & j & k \\
\omega_x' & \omega_y' & \omega_z' \\
x' & y' & z'
\end{vmatrix}
= i(\omega_y'z' - \omega_z'y') + j(\omega_x'z' - \omega_z'x') + k(\omega_x'y' - \omega_y'x')
\]
and performing the dot product results in

\[
(\omega y')^2 - 2(\omega y')(\omega z') + (\omega y')^2 + (\omega x')^2 - 2(\omega x')(\omega z')\]

\[+ (\omega x')^2 + (\omega y')^2 - 2(\omega y')(\omega x') + (\omega x')^2\]

Which upon substituting into kinetic energy equation gives

\[
T = \frac{1}{2} \int \left[ \omega^2 y' (y'^2 + z'^2) + \omega^2 y' (x'^2 + z'^2) + \omega^2 z' (x'^2 + y'^2) + 2\omega x' y' z' - 2\omega x' y' z - 2\omega x' y' y' \right] \, dm
\]

This represents the rotational kinetic energy of one component of a system. Recognizing the definitions of moments and products of inertia and selecting the component coordinate system as the principal axes, we can write

\[
T_{\text{comp}} = \frac{1}{2} \left( I_{xx} + I_{yy} + I_{zz} \right)
\]

This particular selection of coordinates does not affect the final answers because kinetic energy is constant with regards to the coordinate orientation; however, it does simplify the input data and also reduces computer time.

Rotating From Component Coordinates to System Coordinates

At this point we have the kinetic energy about the component axis system. In the computer program it is at this level that mass properties are computed for the preselected shapes such as the cylinder, sphere, etc.

It is necessary to resolve the component mass properties to an axis system parallel to the system coordinates. Once parallel to this system axis then we can translate to the system center of gravity by the usual parallel axis theorem. The derivation is similar to that presented in reference 5.

In deriving the rotational transformation from the component to the system coordinates the expression for kinetic energy is again used. Given a body rotating with an angular velocity \( \vec{\omega} \) we know that its kinetic energy is invariant with regard to the coordinate orientation.
The kinetic energy in the system coordinates is

\[ T_{sys} = \frac{1}{2} \int \left( \mathbf{\omega} \times \mathbf{R} \right) \cdot \left( \mathbf{\omega} \times \mathbf{R} \right) \, dm \]

Defining the products of inertia as

\[ I_{xz} = \int yz \, dm, \quad I_{yz} = \int yz \, dm, \quad I_{xy} = \int xy \, dm \]

the energy equation becomes

\[ T_{sys} = \frac{1}{2} \left[ I_{xx} \omega_x^2 + I_{yy} \omega_y^2 + I_{zz} \omega_z^2 + 2I_{xz} \omega_x \omega_z + 2I_{yz} \omega_y \omega_z + 2I_{xy} \omega_x \omega_y \right] \]

The products of inertia will not necessarily be zero in the system coordinates due to the coordinate rotations and, therefore, the products must be included.
It is necessary to write the angular velocity of the component system coordinates in terms of the system coordinates. For an arbitrary vector it can be written

\[ \vec{\omega} = \vec{\omega}_{comp} = \vec{\omega}_{system} \]

\[ = i'\omega_x' + j'\omega_y' + k'\omega_z' = i\omega_x + j\omega_y + k\omega_z \]

Performing the dot product gives

\[ \omega_x' = i' \cdot i\omega_x + i' \cdot j\omega_y + i' \cdot k\omega_z \]

\[ \omega_y' = j' \cdot i\omega_x + j' \cdot j\omega_y + j' \cdot k\omega_z \]

\[ \omega_z' = k' \cdot i\omega_x + k' \cdot j\omega_y + k' \cdot k\omega_z \]

Recognizing the direction cosines results in

\[ \begin{bmatrix} \omega_x' \\ \omega_y' \\ \omega_z' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \]

Writing the component kinetic energy in matrix algebra

\[ T_{comp} = \frac{1}{2} (\omega')^T [I'] [\omega'] = \frac{1}{2} \left( [DC] (\bar{\omega}) \right)^T [I'] [DC] \{\omega\} \]

where \([DC]\) is the direction cosine matrix shown above. Equating the system and component energy results in

\[ T_{system} = T_{comp} \]

\[ \frac{1}{2} \{\omega\}^T [I] \{\omega\} = \frac{1}{2} \{\omega\}^T [DC]^T [I'] [DC] \{\omega\} \]

and the resulting system inertial matrix is found to be
Expanding the matrix equation we get

\[
[I] = [DC]^T [I'] [DC]
\]

The direction cosines are determined by the method presented in appendix A.

Transferring by Parallel Axis Theorem

Now that the moments of inertia are in a system parallel to the system coordinates, we now translate by the parallel axis theorem

\[
I_{XX} = I_{XXCO} + m[(Y_{co} - \bar{Y})^2 + (Z_{co} - \bar{Z})^2]
\]
\[
I_{YY} = I_{YYCO} + m[(X_{co} - \bar{X})^2 + (Z_{co} - \bar{Z})^2]
\]
\[
I_{ZZ} = I_{ZZCO} + m[(X_{co} - \bar{X})^2 + (Y_{co} - \bar{Y})^2]
\]
\[
I_{XY} = I_{XYCO} + m[(X_{co} - \bar{X})(Y_{co} - \bar{Y})]
\]
\[
I_{XZ} = I_{XZCO} + m[(X_{co} - \bar{X})(Z_{co} - \bar{Z})]
\]
\[
I_{YZ} = I_{YZCO} + m[(Y_{co} - \bar{Y})(Z_{co} - \bar{Z})]
\]
Where $I_{XCO}$, $I_{YCO}$, and $I_{ZCO}$ are the component moments of inertia rotated parallel to the system coordinates, and $X, Y, Z$ are the system center of mass coordinates and $X_{CO}, Y_{CO}, Z_{CO}$ are the coordinates of the component center of mass.

**COMPUTER PROGRAM**

This computer program is written in Fortran IV computer language. All names and descriptions are assigned in the first part of the program. Thirteen sections have been written using 13 common shapes usually found in spacecraft. The program is directed by the input data which singles out the section or shape factor desired to be used through the "go to" statement. The operation of the program is illustrated in figure 1 with a computer flow diagram.

The input data for each item is listed on two data cards. The basic input for each item will vary depending on the shape factor used. Each shape factor with the necessary data is discussed in the input data instructions.

After the data cards are supplied to the program the following operations are performed. The component mass properties are first printed with the moments of inertia about the component axis rotated parallel to the system coordinates. These mass properties are transferred to the system center of gravity and the following are computed: System weight, inertias about the system center of gravity, inertias about the origin, center of gravity of the system and products of inertias of the system. Based on this generated information, inertias about the system principal axes and their location is subsequently computed.

A listing of the computer program is found in appendix B.

**Selection of Coordinate Points**

The selection of points "i" and "j" determines the length of the member as well as the first three direction cosines. Point "k" is required to calculate the other six direction cosines. Shown in figure 2 are the two coordinate systems used in this program. Point "i" locates the system coordinates $(X_i, Y_i, Z_i)$ for the origin of the component axes and point "j" determines the direction of the "x" axis of the component coordinates.

In order to determine the directions of the "y" and "z" axes, point "k" is required. This point can be anywhere in the x-y plane. If it is omitted, then the program automatically positions the "y" axis parallel to the X-Y plane. For a body of revolution point "k" is not required. The following figures (2(a) and 2(b)) describe points i, j, and k.

The main deck is referred to as the computer program without the necessary data. It is always necessary to have a 789 card following the main deck and a 789 card, then a 6789 card following the data. The 6789 card separates one program from another. A description of these cards and their formats follows figure 2.
FIGURE 1 - COMPUTER FLOW DIAGRAM
LOCATION OF POINTS "i" AND 'j"

Figure 2(a)

LOCATION OF POINTS "k"

Figure 2(b)
### 1st Data Card

<table>
<thead>
<tr>
<th>Item No.</th>
<th>Format</th>
<th>Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>I3</td>
<td>I3 format</td>
<td>1 through 3</td>
</tr>
<tr>
<td>Description</td>
<td>2A9 format</td>
<td>4 through 21</td>
</tr>
<tr>
<td>Shape</td>
<td>I2 format</td>
<td>22 and 23</td>
</tr>
<tr>
<td>Weight or density</td>
<td>F9.4 format</td>
<td>24 through 32</td>
</tr>
<tr>
<td>A</td>
<td>F8.3 format</td>
<td>33 through 40</td>
</tr>
<tr>
<td>B</td>
<td>F8.3 format</td>
<td>41 through 48</td>
</tr>
<tr>
<td>C</td>
<td>F8.3 format</td>
<td>49 through 56</td>
</tr>
<tr>
<td>D</td>
<td>F8.3 format</td>
<td>57 through 64</td>
</tr>
<tr>
<td>F</td>
<td>F8.3 format</td>
<td>65 through 72</td>
</tr>
</tbody>
</table>

It is to be noted that the input data variables A, B, C, D, and F can be geometric dimensions, cross-sectional areas, area moments of inertia, and mass moment of inertia.

### 2nd Data Card

| XI       | F8.3 format| 1 through 8 |
| YI       | F8.3 format| 9 through 16 |
| ZI       | F8.3 format| 17 through 24 |
| XJ       | F8.3 format| 25 through 32 |
| YJ       | F8.3 format| 33 through 40 |
| ZJ       | F8.3 format| 41 through 48 |
| XK       | F8.3 format| 49 through 56 |
| YK       | F8.3 format| 57 through 64 |
| ZK       | F8.3 format| 65 through 72 |

**ORIGINAL PAGE IS OF POOR QUALITY**
ORDER OF CARDS FOR COMPUTER PROGRAM

6789 CARD

789 CARD

DATA

789 CARD

MAIN DECK

DATA CARDS AND FORMATS
(INPUT DATA FOR EACH ITEM)

<table>
<thead>
<tr>
<th>ITEM NO.</th>
<th>DESCRIPTION</th>
<th>DF OR DENSITY</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1ST DATA CARD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2ND DATA CARD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ORIGINAL PAGE IS OF POOR QUALITY
Component shapes in the existing programs are:

- Shape 1: Discrete Mass
- Shape 2: Cylinder
- Shape 3: Truncated Cone
- Shape 4: Torus
- Shape 5: Beam (Arbitrary Cross Section)
- Shape 6: Circular Road (Arbitrary Cross Section)
- Shape 7: Spherical Segment
- Shape 8: Sphere
- Shape 9: Hemisphere
- Shape 10: Parallelepiped
- Shape 11: Swept Trapezoidal Panel
- Shape 12: Symmetric Swept Trapezoidal Panels
- Shape 13: Curved Rectangular Panel

This program can easily be modified to include additional shapes.

Frequently where precise weights of components are known it is more convenient to input this weight rather than an average density which would have to be calculated. The value of .4 has been chosen as the limiting value for inputing Rho as density (ib/cu in). A value greater than .4 is used as total weight (lbs). In some cases referred to as "thin wall" the total weight of the component must be input. The degree of flexibility for inputing various shapes can be determined from the "shape data input instructions".

The program can be used to determine the mass properties of a component with hollows or voids. The component is treated as a standard shape with the hollows or voids included as solid material. The hollows or voids are then input as standard shapes having negative values for weight or density. The program will compute the actual weight, center of gravity, and moments and products of inertia of the component.

In the following programs for a variety of shapes the inertias about the x, y and z component axes are represented respectively by computer symbols IXXCG, IYYCG and IZZCG. For some of the shapes expressions for the moments of inertia are given; however, for the more complex shapes they are omitted but can be found by referring to the computer program in Appendix B.

Note that component axes have been located so the x axis is a principal axis and the y and z axes either coincide with, or are parallel to principal axes. Any other than the aforementioned x, y, and z axes locations will incur an error in the main computer program with shape no. 11 being the sole exception. In this instance, the x, y, and z axes locations were chosen for convenience in accordance with the "shape data input instruction" and the x axis is rotated to a principal axis by the program.
SHAPE DATA INPUT INSTRUCTION

SHAPE 1
DISCRETE MASS

Center of gravity

Input Data: Item, Description, Shape, RHO (wt.)
A(Ixx), B(Iyy), C(Izz)
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

OPTIONS AVAILABLE
No.1 Input data just as indicated above.
No.2 If negligible, inertias A(Ixx), B(Iyy) and C(Izz) may be omitted, in which case points j and k are not required.

NOTES: Never locate point j at the system origin and if input, A(Ixx), B(Iyy) and C(Izz) must be inertias about principal axes in slug-ft.

SHAPE 2
CYLINDER

Input Data: Item, Description, Shape, RHO (density)
A, B, C (option No. 2 only)
XI, YI, ZI, XJ, YJ, ZJ

OPTIONS AVAILABLE
No.1 Input data just as indicated above.
No.2 Cylinder may be segmented requiring a C value be input.
No.3 Total weight may be input for RHO in both previous options but only if it is input more than .4 pounds.
No.4 Input A and B equal and total weight for RHO and program treats shape as a thin-wall cylinder.

NOTES: Point k is not required with any option and density as such must never be input more than .4 pounds per cu. in.
SHAPE DATA INPUT INSTRUCTION

SHAPE 3
TRUNCATED CONE

Input Data: Item, Description, Shape, RHO (density), A, B, C, D
XI, YI, ZI, XJ, YJ, ZJ

OPTIONS AVAILABLE
No. 1 Input data just as indicated above
No. 2 A total weight may be input for RHO but only if it is more than .4 pounds.

NOTES: The values of A minus C and or B minus D must never equal zero.
Point i is always at the cones larger end and point k is not required.
Density as such must never be input greater than .4 lbs./cu.in.

SHAPE 3 (SPECIAL CASE)
THIN-WALL TRUNCATED CONE

Input Data: Item, Description, Shape
RHO (Density), A, B, C, F
XI, YI, ZI, XJ, YJ, ZJ

NOTES: In this case program assumes all mass is concentrated midway between inner and outer surfaces.
Point i is always at the cone's larger end and point x is not required.
SHAPE DATA INPUT INSTRUCTION

SHAPE 4
TORUS

\[ \begin{align*}
\text{Input Data: } & \text{ Item, Description, Shape, RHO (density), } A, D \\
& \text{ XI, YI, ZI, XJ, YJ, ZJ} \\
\text{NOTE: } & \text{ Point "k" is not required.}
\end{align*} \]

SHAPE 5
BEAM
(ARBITRARY CROSS SECTION)

\[ \begin{align*}
\text{Input Data: } & \text{ Item, Description, Shape, RHO (density), } A \text{ (area), } \\
& \text{ B(Iyy), C(Izz) (Area moments of inertia in inches}^4) \\
& \text{ XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK} \\
\text{NOTES: } & \text{ If beam is a body of revolution about the x (centroidal) axis, point k is not required. Iyy and Izz are area moments of inertia about principal axes of the beam cross section taken in a plane normal to x axis.}
\end{align*} \]
SHAPE DATA INPUT INSTRUCTION

SHAPE 6
CIRCULAR ROD
(ARBITRARY CROSS SECTION)

Input Data:
Item, Description, Shape, RHO (wt), A, B (Area "C" Ix'x'), C
XI, YI, ZI, XJ, YJ, ZJ
(Input Area "C" Ix'x' in inches²)

NOTES:
Point "k" is not required and weight instead of density is the required input for RHO.
The solution for this shape is approximate in that it may be as much as one % less than correct if rod dimensions t and w are as much as 25% of dimension A. (Error incurred tends to increase as this % increases)

SHAPE ~
SOLID SEGMENT

Input Data:
Item, Description, Shape, RHO (density), B, C (outside radius)
XI, YI, ZI, YJ, YJ, ZJ

OPTIONS AVAILABLE
No. 1 Input data just as indicated above
No. 2 Total weight may be input for RHO if more than .4 lb.

NOTES:
Point k is not required. When computing as a solid spherical segment, B dim. will become equal the distance between points "i" and "j". Density as such must never be input greater than .4 lbs./cu. in.
SHAPE DATA INPUT INSTRUCTION

SHAPE 8
SPHERE

\[
\begin{align*}
\text{DXCG} &= \text{XII}-\text{XI2} \\
\text{IYCG} &= \text{IXXCG} \\
\text{IZZCG} &= \text{IXXCG}
\end{align*}
\]

Input Data: Item, Description, Shape, RHO (density), A,B
\[\text{XI, YI, ZI}\]

NOTE: Points "j" and "k" are not required. If B = A the program selects thin-wall equations; therefore, weight instead of density should be input for RHO.

SHAPE 9
HEMISPHERE

\[
\begin{align*}
\text{IXXCG} &= \text{XII}-\text{XI2} \\
\text{IYCG} &= \text{XI3}-\text{XI4} \\
\text{IZZCG} &= \text{IYCG}
\end{align*}
\]

Input Data: Item, Description, Shape, RHO (density), A
\[\text{XI, YI, ZI, XJ, YJ, ZJ}\]

NOTES: Point "k" is not required. If A = Lgth (i,j) the program selects thin-wall equations; therefore, weight instead of density should be input for RHO.
SHAPE DATA INPUT INSTRUCTION

SHAPE 10
PARALLELEPIPED

Point k
(Can lie anywhere in the xy plane)

Input Data:
- Item, Description, Shape, RHO (density), A, B, C, D, F
- XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

OPTIONS AVAILABLE
- No. 1 Input data just as indicated above.
- No. 2 A total weight may be input for RHO but only if it is more than .4 pounds.
- No. 3 Input D equal A and the program selects thin-wall equations therefore total weight must be input for RHO.

NOTE: Density as such must never be input greater than .4 lbs./cu.in.
SHAPE DATA INPUT INSTRUCTION

SHAPE 11
SWEPT TRAPEZOIDAL PANEL
(THICK WALL)

Input Data:
Item, Description, Shape, RHO (density), A, B, C, F
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK
(Assign $i$ to $C$ according to sweep of panel)

Formulas designed to facilitate input data determination:

\[ a = \frac{C(B)}{B-F} \]
\[ b = \frac{C(B+2F)}{3(B+F)} \]
\[ c = \frac{Lenth(B+2F)}{3(B+F)} \]

OPTIONS AVAILABLE
No.1 Input data just as indicated above.
No.2 Total weight may be input for RHO if more than .4 lbs.

SHAPE 11 (SPECIAL CASE)
UNSWEPT TRAPEZOIDAL PANEL
(THICK WALL)

Input Data:
Item, Description, Shape, RHO (density), A, B, F
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

OPTIONS AVAILABLE
No.1 Input data just as indicated above.
No.2 Total weight may be input for RHO if more than .4 lbs.

NOTES:
Dimension F must never be more than 98% of B.
Density as such must never be input greater than .4 lb./cu.in.

Point k (Can lie anywhere on the $y$ axis except at point i )

NOTES:
Dimension F must never be more than 95% of B.
Density as such must never be input greater than .4 lb./cu.in.

ORIGINAL PAGE IS OF POOR QUALITY
SHAPE DATA INPUT INSTRUCTION

SHAPE 12
SYMMETRIC SWEPT TRAPEZOIDAL PANELS (THICK WALL)

Point k
can lie anywhere
in the xy plane

Input Data:
Item, Description, Shape, RHO (density), A, B, C, D, F
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK
(Assign 2 to C according to sweep of panel)

OPTIONS AVAILABLE
No. 1 Input data just as indicated above.
No. 2 Total weight may be input for RHO
if it is more than 0.4 lb.

Formulas designed to facilitate
input data determination:

\[
b = \frac{C(B+2F)}{3(B+F)}
\]
\[
c = \frac{\text{Lgth}(B+2F)}{3(B+F)}
\]

NOTES:
Dimension F must be no more than 98% of B.
Density as such must never be input greater
than 0.4 lb./cu.in.

SHAPE 13
CURVED RECTANGULAR PANEL
(THIN WALL)

Point k
can lie anywhere
in the xy plane

Input Data:
Item, Description, Shape
RHO (density), A, B, C
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

NOTE: Plane xy is one about which symmetry exists.

22
EXAMPLE PROBLEMS

Two example problems are presented in order to show required input data. These problems were selected due to the simple calculations involved and thus could be checked by hand calculations.

Example Problem 1

Problem 1 was taken from reference 3. It is a cylinder skewed in the y-z plane. The moments of inertia about the system origins are given in this reference and are used for comparison in this paper. It should be noted that "g" (acceleration of gravity) was taken to be 32.0 ft/sec² in this reference rather than 32.2 ft/sec² (386 in./sec²). Depending on the value of "g" selected by the program user, the term "cons" has to be changed accordingly.

The axis through the center of the component must always be the x-axis for the computer program. Therefore, the y₃ axis of this problem corresponds with the x-axis of the computer program; the y₁ axis corresponds with the y-axis and the y₂ axis corresponds with the z-axis.

Find moments of inertia I'₁₁, I'₂₂, I'₃₃ which corresponds to Iₓₓ₀, Iᵧᵧ₀, Izz₀ in our coordinate system.

Given: W = 1 slug = 32 lb
R = 24 inches
L = 36 inches
θ = cos⁻¹ 3/5 = 53° 8'

Solution: (a) Direction cosines for the transformation are

\[
\mathbf{a}_{ij} = \begin{bmatrix}
1 & 0 & 0 \\
0 & 2/5 & 4/5 \\
0 & -4/5 & 3/5
\end{bmatrix}
\]

And numerical values of the Iᵢⱼ are

\[
I_{ij} = \begin{bmatrix}
4 & 0 & 0 \\
0 & 4 & 0 \\
0 & 0 & 2
\end{bmatrix}
\]
The results of problem 1 as computed in reference 3 are now given. Hence, from the transformation equations

**System Moments of Inertia**

\[ I'_{11} = a_{11}^{11} I_{11} + a_{12}^{12} I_{22} + a_{13}^{13} I_{33} + 2a_{12}^{12} a_{13}^{13} I_{23} + 2a_{11}^{11} a_{13}^{13} I_{13} + 2a_{11}^{11} a_{12}^{12} I_{12} \]

\[ I'_{xx} = (1)^2 4 + 0 + 0 + 0 + 0 = 4 = \text{slug ft}^2 \quad (I'_{xx} = I'_{11}) \]

\[ I'_{22} = a_{21}^{21} a_{11}^{11} + a_{22}^{22} a_{22}^{22} + a_{23}^{23} a_{23}^{23} I_{33} + 2a_{22}^{22} a_{23}^{23} I_{23} + 2a_{21}^{21} a_{23}^{23} I_{13} + 2a_{21}^{21} a_{22}^{22} I_{12} \]

\[ I'_{yy} = 0 + (\frac{2}{3})^2 4 + (\frac{4}{3})^2 2 + 0 + 0 + 0 = \frac{68}{25} = 2.72 \text{ slug ft}^2 \quad (I'_{yy} = I'_{22}) \]

\[ I'_{zz} = a_{31}^{31} a_{11}^{11} + a_{32}^{32} a_{32}^{32} I_{22} + a_{33}^{33} a_{33}^{33} I_{33} + 2a_{32}^{32} a_{33}^{33} I_{23} + 2a_{31}^{31} a_{33}^{33} I_{13} + 2a_{31}^{31} a_{32}^{32} I_{12} \]

\[ I'_{zz} = 0 + - \frac{4}{3} 4 + \frac{3}{5} 2 + 0 + 0 + 0 = \frac{32}{25} = 3.28 \text{ slug ft}^2 \]

This problem is now computed using Computer Program NDC77C2. Shown below are the coordinates of points \( i, j, \text{ and } k \) for problem 1.

**Point "i"**

\[ \begin{align*}
X_i &= 0 \\
Y_i &= 0 \\
Z_i &= 0
\end{align*} \]

**Point "j"**

\[ \begin{align*}
Y_j &= 30 \sin 55^\circ \text{ ft} = 23.3 \\
Z_j &= 30 \cos 55^\circ \text{ ft} = 21.6
\end{align*} \]

**Point "k"**

Omit since not required for cylinder

Reference is now made to the necessary data cards to compute this problem.
<table>
<thead>
<tr>
<th>ITEM NO.</th>
<th>SHAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>WT. OR DENSITY</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>CYLINDER TEST</td>
<td>32.0</td>
<td>24</td>
</tr>
</tbody>
</table>

---

**FORTRAN STATEMENT**

1ST DATA CARD

```
YJ   ZJ
28.8 21.8
```

2ND DATA CARD

```
```
We now have the results of problem 1 using Computer Program
(Note that the gravitational constant used here was 32.166... rather than 32.0 used in ref.)

<table>
<thead>
<tr>
<th>INPUT DATA LISTED BELOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COMPONENT DATA LISTED BELOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SYSTEM DATA LISTED BELOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Example Problem 2

Problem 2 was taken from reference 2. In this space structure, the weight has been 'lumped' or concentrated at the joints. In the program being presented, this is not required but is used here only for illustration. To compare moment of inertia, the numbers given in reference 2 should be converted to slug-ft².

Ixx = Iyy - \( \frac{22038.464}{4.608E3} \) = 4.783 slug-ft²

Izz = 9000.00 \( \frac{4.0608F3}{4.0608F3} \) = 1.953 slug-ft²

ORIGINAL PAGE IS OF POOR QUALITY
Example Problem 2

INPUT
MEMBER AREAS - 0.01 in^2
WEIGHT AT EACH JOINT - 150 lb
MODULUS OF ELASTICITY - 10^12 psi
SEE THE FIRST TWO PAGES OF THE SAMPLE
PROBLEM FOR DETAILS OF THE INPUT

<table>
<thead>
<tr>
<th>JOINT COORDINATES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
</tbody>
</table>

The results of problem 2 as computed in reference 6 are now given.

This problem is now computed using Computer Program ND0702
| ITEM | DESCRIPTION | F1 | IZICO | IZICO | IZICO | IZICO | IZICO | IZICO | IZICO | IZICO | IZICO | IZICO | IZICO |
|------|-------------|----|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1    | NODE 1      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 2    | NODE 2      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 3    | NODE 3      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 4    | NODE 4      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 5    | NODE 5      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 6    | NODE 6      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 7    | NODE 7      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 8    | NODE 8      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 9    | NODE 9      | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 10   | NODE 10     | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 11   | NODE 11     | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 12   | NODE 12     | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 13   | NODE 13     | 15.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

**System Data Listed Below (Torques, Inertias, and Velocities)**

- **Input Torque:** 1.0000
- **Output Torque:** 1.0000
- **Rotation:** 0.0000
- **Output Velocity:** 0.0000
- **Inertias:** 0.0000

Note: No principal axes definition is included since they will lie on one of symmetry.
EXAMPLE PROBLEM 3
A COMPOSITE STRUCTURE
(\frac{1}{5} SCALE)

Item No. 1
(1.5 lb. total wt.)

Sta. 35.0

Item No. 2
(1.5 in. thick)

Sta. 25.0

Item No. 3

Item No. 6
(Cutout in Cylinder)

Sta. 11.0

Item No. 7
(5 lb. wiring etc. evenly distributed inside cylinder)

Sta. 0.0

Item No. 4
(.5 in. thick)

Sta. -5.0

NOTES:
Points i and j shown on the structure are those assumed when data was input to the inertia program. Also, some salient dimensions are included on the components to assist in relating component data to the computer input. Except for items No. 6 and No. 7, density is input for RHO.

The computer output relative to example problem 3 follows and includes the data input, the computer calculated component data, the summed data and lastly, the inertias about the system principal axes and the individual axis locations.

ORIGINAL PAGE IS OF POOR QUALITY
<table>
<thead>
<tr>
<th>ITEM</th>
<th>DESCRIPTION</th>
<th>SHAPE</th>
<th>RND</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ANTENNA</td>
<td>7</td>
<td>1900000000.0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15.000</td>
<td>-0.000</td>
<td>0.000</td>
<td>20.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>DIELECTRIC</td>
<td>11</td>
<td>-1500000000.0</td>
<td>1.000</td>
<td>0.000</td>
<td>-2.000</td>
<td>0.000</td>
<td>3.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>Cyl</td>
<td></td>
<td></td>
<td>15.000</td>
<td>0.000</td>
<td>0.000</td>
<td>21.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>HOE CYL</td>
<td>11</td>
<td>-1000000000.0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15.000</td>
<td>0.000</td>
<td>0.000</td>
<td>20.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>VIT</td>
<td></td>
<td>-2000000000.0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>5</td>
<td>Cyl Seg Void</td>
<td>15</td>
<td>-1000000000.0</td>
<td>2.000</td>
<td>0.000</td>
<td>1.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15.000</td>
<td>0.000</td>
<td>0.000</td>
<td>20.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>6</td>
<td>WIRING ETC VOL</td>
<td>2</td>
<td>-5000000000.0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

**Component Data Listed Below**

<table>
<thead>
<tr>
<th>ITEM</th>
<th>DESCRIPTION</th>
<th>Vol.</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ANTENNA</td>
<td></td>
<td>1.9800</td>
<td>.0084</td>
<td>.0060</td>
<td>.0226</td>
<td>.0097</td>
<td>.0077</td>
<td>.0044</td>
</tr>
<tr>
<td>2</td>
<td>DIELECTRIC</td>
<td></td>
<td>16.2126</td>
<td>.0184</td>
<td>.0040</td>
<td>.0534</td>
<td>.0010</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td>3</td>
<td>HOE CYL</td>
<td></td>
<td>21.0010</td>
<td>.0274</td>
<td>.1294</td>
<td>.1677</td>
<td>.2307</td>
<td>.223</td>
<td>.0000</td>
</tr>
<tr>
<td>4</td>
<td>VIT</td>
<td></td>
<td>2.6400</td>
<td>.0074</td>
<td>.0028</td>
<td>.0028</td>
<td>.0000</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td>5</td>
<td>Cyl Seg Void</td>
<td></td>
<td>14.7110</td>
<td>.0067</td>
<td>.0050</td>
<td>.0049</td>
<td>.0000</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td>6</td>
<td>WIRING ETC VOL</td>
<td></td>
<td>21.0000</td>
<td>.0127</td>
<td>.0091</td>
<td>.0091</td>
<td>.0000</td>
<td>.0000</td>
<td>.0000</td>
</tr>
</tbody>
</table>

**System Data Listed Below**

<table>
<thead>
<tr>
<th>SYS</th>
<th>INT</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
<th>FYDC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>53.000</td>
<td>.204</td>
<td>.040</td>
<td>.104</td>
<td>.168</td>
<td>12.177</td>
<td>.287</td>
<td>.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Inertia (Eigenvalues) about System Principal Axes with Axis Direction Cosines (Eigenvectors) Relating the Principal Axes to the x, y, and z System Axes in that Sequence**

Eigenvalue(1) = 0.20760E+00

Eigenvector(1) = 0.99999E+00 0.00000E+00 0.00000E+00

Eigenvalue(2) = 0.23760E+01

Eigenvector(2) = 0.99999E+00 0.00000E+00 0.00000E+00

Eigenvalue(3) = 0.12420E+01

Eigenvector(3) = 0.99999E+00 0.00000E+00 0.00000E+00

Eigenvalue(4) = 0.24510E+01

Eigenvector(4) = 0.00000E+00 0.00000E+00 0.10000E+01

30
**CONVERSION TO INTERNATIONAL SYSTEM OF UNITS**

<table>
<thead>
<tr>
<th>CONVERT FROM</th>
<th>TO</th>
<th>MULTIPLY BY</th>
</tr>
</thead>
<tbody>
<tr>
<td>INCHES</td>
<td>METERS</td>
<td>.025 400</td>
</tr>
<tr>
<td>INCHES&lt;sup&gt;2&lt;/sup&gt;</td>
<td>METERS&lt;sup&gt;2&lt;/sup&gt;</td>
<td>.000 645 160</td>
</tr>
<tr>
<td>FEET</td>
<td>METERS</td>
<td>.304 800</td>
</tr>
<tr>
<td>FEET&lt;sup&gt;2&lt;/sup&gt;</td>
<td>METERS&lt;sup&gt;2&lt;/sup&gt;</td>
<td>.092 903 040</td>
</tr>
<tr>
<td>POUNDS</td>
<td>KILOGRAMS</td>
<td>.153 492 370</td>
</tr>
<tr>
<td>POUNDS/INCHES&lt;sup&gt;3&lt;/sup&gt;</td>
<td>KILOGRAM/METER&lt;sup&gt;3&lt;/sup&gt;</td>
<td>27 679.905 ....</td>
</tr>
<tr>
<td>POUNDS/FEET&lt;sup&gt;3&lt;/sup&gt;</td>
<td>KILOGRAM/METER&lt;sup&gt;3&lt;/sup&gt;</td>
<td>16.018 463 ....</td>
</tr>
<tr>
<td>SLUG</td>
<td>KILOGRAMS</td>
<td>14.593 9029 ....</td>
</tr>
<tr>
<td>SLUG FEET&lt;sup&gt;2&lt;/sup&gt;</td>
<td>KILOGRAM METER&lt;sup&gt;2&lt;/sup&gt;</td>
<td>1.355 8179 ....</td>
</tr>
</tbody>
</table>
CONCLUSIONS

A computer program for determining mass properties of a rigid structure is presented. The structure is broken down into preselected shapes with known properties, and input data are supplied to completely describe each shape. For complicated structures skewed in space, this program offers a practical solution to a tedious and time-consuming task. It is also practical to use this program for problems that involve repetitious or lengthy calculations.
REFERENCES

APPENDIX A

DIRECTION COSINE DERIVATION

The direction cosines are used to transform the properties from the component coordinate system \( x, y, z \) to system coordinates \( X, Y, Z \). Three points \((i, j, k)\) shown in the figure below, define the vectors \( \vec{V}_1 \) and \( \vec{V}_2 \).

Vector \( \vec{V}_1 \) is arbitrarily selected to be coincident with the "x" axis. A unit vector on this axis can be written

\[
\vec{1}_x = \frac{\vec{V}_1}{|\vec{V}_1|}
\]

Taking the vector cross product of \( \vec{V}_1 \) with \( \vec{V}_2 \) and dividing by the resulting magnitude gives a unit vector on the "z" axis

\[
\vec{1}_z = \frac{\vec{V}_1 \times \vec{V}_2}{|\vec{V}_2|}
\]

Similarly, a unit vector on the "y" axis is found from

\[
\vec{1}_y = \vec{1}_z \times \vec{1}_x
\]

The direction cosines are the \( X, Y, Z \) components of the unit vectors on the \( x, y, z \) axes.
The direction cosine for the x axis is written as

\[
\begin{align*}
LX &= (XJ - XI) / LGTH \\
MX &= (YJ - YI) / LGTH \\
NX &= (ZJ - ZI) / LGTH
\end{align*}
\]

where the length is

\[
LGTH = \sqrt{((XJ - XI)^2 + (YJ - YI)^2 + (ZJ - ZI)^2)}
\]

The vector \( \mathbf{V}_2 \) can be written

\[
\begin{align*}
T1 &= XK - XI \\
T2 &= YK - YI \\
T3 &= ZK - ZI
\end{align*}
\]

A vector on the z axis is found by taking the vector cross product of \( \mathbf{1}_x \) and \( \mathbf{V}_2 \)

\[
\begin{align*}
LZ &= MX \times T3 - T2 \times NX \\
MZ &= NX \times T1 - T3 \times LX \\
NZ &= T2 \times LX - T1 \times MX
\end{align*}
\]

The length is

\[
T4 = \sqrt{(LZ^2 + MZ^2 + NZ^2)}
\]

Normalizing to get a unit vector

\[
\begin{align*}
LZ &= LZ / T4 \\
MZ &= MZ / T4 \\
NZ &= NZ / T4
\end{align*}
\]

The unit vector on the y axis has a magnitude of one and is determined by the vector cross product \( \mathbf{1}_z \) and \( \mathbf{1}_x \)

\[
\begin{align*}
LY &= MZ \times NX - MX \times NZ \\
MY &= NZ \times LX - NX \times LZ \\
NY &= MX \times LZ - LX \times MZ
\end{align*}
\]

Writing the nine terms in matrix form, we get

\[
[DC] = \begin{bmatrix}
LX & MX & NX \\
LY & MY & NY \\
LZ & MZ & NZ
\end{bmatrix}
\]

35
APPENDIX 8

PROGRAM NAME (INPUT,OUTPUT,TAPES=INPUT,TAPES6=OUTPUT)

C LATEST DIRGOS IN MAIN DECK AND 13 SHAPES 12/20/72

DIMENSION ITEM(200),RH0(200),A(200),R(200),C(200),
1 N(200),F(200),XI(200),YI(200),ZI(200),XJ(200),YJ(200),ZJ(200),
1 XK(200),YK(200),ZK(200),IXXC(200),IXYCYG(200),IXZCG(200),IXYCG(
1200),IXYCG(200),IXZCG(200),XCG(200),YC(200),ZC(200),XL(200),
1 YL(200),ZL(200),IXXC1(200),IXYCYG1(200),IXZCG1(200),IXZCG(200),
1 YZC1(200),YZC2(200),W(200),N(200),P(200),ARR(3,3),F(3),CRW(3),
1 REAL IXX,IXY,IXZ,IXXC,IXYC,IXZC,IXYG,IXYCG,IXZCG,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC.
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC.
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
1 IXZC,IXYCG,IXZC,IXXC,IXYC,IXZC,IXXC,IXYC,IXZC,
DISCRETE MASS *SHAPE 1*
POINT J MUST BE SELECTED ANYWHERE ON THE X AXIS FOR DIR. COSINES

1 = (1) = M(I)
IXCG(I) = A(I) * CON3
IYCG(I) = B(I) * CONS
ITZCG(I) = C(I) * CONS
IFS = (XJ(I), YJ(I), ZJ(I)) * CON3
IF = (ARS(YJ(I)), HE, OB) OR. TANS(ZJ(I)), HE, 0
)
1)) GO TO 704
LGT = 10
XJ(I) = 1 * XI(I) + 10
IF = (XJ(I), YJ(I), ZJ(I)) * XI(I) + 10
704 XL(I) = 0
L(I) = T(I)
600

CYLINDER # SHAPE 2#

2 IF = (A(I), B(I), C(I)) GO TO 33
IF = (X(I), Y(I), Z(I)) R(I) = 0
IF = (C(I), LT, OB) C(I) = 0
VOLCYL = PT * A(I) * R(I) * (LGT - C(I))
A(I) = R(O) * C(I) * VOLCYL
IF = (RHO(I), GT, OB) R(I) = RHO(I)
R(I) = (T(VOLCYL)
IXCG(I) = 5 * A(I) * (A(I) * R(I) * R(I))
IYCG(I) = 25 * A(I) * (A(I) * R(I) * R(I)) * (LGT - 3 * C(I))
5/(12, * (LGT - C(I)))
ITZCG(I) = T(VOLCYL)
XL(I) = LGT / 2
GO TO 60
CYLINDER (THIN WALL) *SHAPE 2*

33 *(I)*RH0(I)
XL(I) = LGTH/2.
ITXCG(I) = (RH0(I)*A(I)**2)
IYYCG(I) = 5*RH0(I)*A(I)**2+RH0(I)*(LGTH**3=C(I)**3)/
2(LGTH=C(I)**12.0)
ITZCG(I) = IYYCG(I)
GO TO 60

TRUNCATED CONE *SHAPE 3* (REF=R. HULL)

3 IF (B(I)=EQ.A(I)) GO TO 30
* (I) = RH0(I)*VOL
IF (RH0(I)<GT.0) *(I) = RH0(I)
RH0(I) = *(I)/VOL
XL(I) = (LGTH**2)*(A(I)**2+2*A(I)*C(I)+C(I)**2+B(I)**2+B(I)*C(I))
*D(I) = 3*O(I)**2*2.618/VOL
* (I) = 0.3*1.5153*RH0(I)*((C(I)**5-A(I)**5)*LGTH/(C(I)+A(I))= 
$O(I)**5=B(I)**5)*LGTH/(D(I)+B(I))
* (I) = 5.1465*RH0(I)*LGTH**3*( (A(I)**2+B(I)**2)/3.0 = * (I)
*R(I)**4-A(I)**2+B(I)**2+2*A(I)*C(I)**2+B(I)**2+2*(B(I)=C(I))**2)
* (I) = 1.0472*(LGTH***(A(I)**2+B(I)**2+C(I)**2+B(I)**2+2*(B(I)=C(I))**2)
* (I) = 2.0*RH0(I)*XL(I)**2
ITXCG(I) = 2.0*(X(I)
* (I) = C(I)**2+O(I)**2+1
IZZCG(I) = IYYCG(I)
GO TO 60

TRUNCATED CONE (THIN WALL) *SHAPE 3*

30 VOL = (I)**SRT((A(I)**2+LGTH**2)**3.14159*(A(I)+C(I))
* (I) = VOL/RH0(I)
XL(I) = LGTH/3.0*(2*C(I)+A(I))/((C(I)+A(I))
IYYCG(I) = RH0(I)*((A(I)**2+C(I)**2)+RH0(I)*LGTH**2+RA**1*(1.02*A(I)**2+C(I)**2)/
1.0)*C(I)**2+A(I)**2+2+B(I)**2+2*(B(I)=C(I))**2)
IZZCG(I) = IYYCG(I)
ITXCG(I) = RH0(I)/2.0*A(I)**2+C(I)**2)
GO TO 60

TORUS *SHAPE 4*

A(I) = A(I)+LGH
IF (C(I)<LT.0) D(I)=0,
VOL = 2.0*(LGH**2)**3+2**A(I)
VOL = VOL/2.0*(LGH**2)**3+2**A(I)
ACVOL = VOL
* (I) = RH0(I)+ACVOL
VOL = VOL*VOL
38
\[ \begin{align*}
X1 &= 125X4.1 * (4 \cdot A(I) * 2 + 5 \cdot LGTH * A(I)) \\
X2 &= 125X4.2 * (4 \cdot A(I) * 2 + 5 \cdot D(I) * 2) \\
IYYCG(I) &= (X1 - X2) \\
IZZCG(I) &= IYYCG(I) \\
Y1 &= 25X4.1 * (4 \cdot A(I) * 2 + 3 \cdot LGTH * 2) \\
Y2 &= 25X4.2 * (4 \cdot A(I) * 2 + 3 \cdot D(I) * 2) \\
IXXCG(I) &= Y1 - Y2
\end{align*} \]

GO TO 60

C \ BEAM (ARBITRARY CROSS SECTION) \ SHAPE 5

5 \[ \begin{align*}
XL(I) &= LGTH / 2. \\
IXXCG(I) &= \{(RHO(I) * LGTH) * (B(I) + C(I)) \} \\
IYYCG(I) &= \{(RHO(I) * (A(I) * LGTH + 0.033 * A(I) * LGTH * 3)) \} \\
IZZCG(I) &= \{(RHO(I) * (C(I) * LGTH + 0.033 * A(I) * LGTH * 3)) \} \\
VOL &= A(I) * LGTH \\
* (I) &= RHO(I) * VOL \\
GO TO 60
\end{align*} \]

C \ CIRCULAR ROD (ARBITRARY CROSS SECTION) \ SHAPE 6

6 \[ \begin{align*}
*(I) &= RHO(I) \\
RSCG &= (A(I) * 2 + (B(I) / (4 * A(I) * C(I)))) \\
IXXCG(I) &= RSCG * P * (I) \\
IYYCG(I) &= 5 * RSCG * P * (I) \\
IZZCG(I) &= IYYCG(I) \\
GO TO 50
\end{align*} \]

C \ SPHERICAL SEGMENT \ SHAPE 7

7 \[ \begin{align*}
F(I) &= C(I) * A(I) \\
*(I) &= LGTH \\
G &= LGTH * A(I) \\
VOL &= 1. * 4 / 3 * LGTH * 2 * (3 * C(I) = LGTH) \\
VOL &= 2. * 4 / 3 * LGTH \\
A CVL &= VOL * VOL \\
* (I) &= VOL * RHO(I) \\
IF &= (I) * LGTH * B(I) * 4. \\
* (I) &= RHO(I) \\
RHO(I) &= * (I) / ACVL \\
XBAR &= 75 * (2 * C(I) = LGTH) * 2 / (3 * C(I) = LGTH) \\
XBAR &= 75 * (P * F(I) = G) * 2 / (3 * F(I) = G) \\
X(I) &= (XBAR * 1 * VOL) / XBAR * 2 * VOL / ACVL \\
XL(I) &= X(I) \\
X &= VOL * RHO(I) \\
X &= 2 * 2 * RHO(I) \\
* (I) &= RHO(I) \\
* (I) &= LGTH * 1 * 5 * LGTH
\end{align*} \]

ORIGINAL PAGE IS OF POOR QUALITY
ACVOL = VOL1 * VOL2
W(I) = RHO(I) * ACVOL
XBAR1 = 375 * LGTH
XBAR2 = 375 * A(I)
XL(I) = (XBAR1 * VOL1 * XBAR2 * VOL2) / ACVOL
XM1 = RHO(I) * VOL1
XM2 = RHO(I) * VOL2
XII(I) = 0.4 * XM1 * LGTH**2
XIZ(I) = 0.4 * XM2 * A(I)**2
IXCG(I) = (XI1(I) * XI2(I))
XI3 = 0.26 * XM1 * LGTH**2
XI4 = 0.26 * XM2 * A(I)**2
IYCG(I) = (XI3 * XI4)
IZZCG(I) = IYCG(I)
GO TO 60

C HEMISPHERE (THIN WALL) *SHAPE 9*

35 A(I) = RHO(I)
XL(I) = LGTH/2
IXCG(I) = 0.666 * RHO(I) * LGTH**2
IYCG(I) = 0.166 * RHO(I) * LGTH**2
IZZCG(I) = IYCG(I)
GO TO 40

C PARALLELEPIPED *SHAPE 1*

10 IF(O(I), EQ, A(I))) GO TO 36
IF(O(I), LT, 0) O(I) = 0,
VOL1 = LGTH * A(1) * A(I)
VOL2 = C(I) * F(I) * O(I)
ACVOL = VOL1 * VOL2
X(I) = X(I) + ACVOL
P(I) = -(P(I)) = X(I) / ACVOL
XL(I) = LGTH/2
XM1 = VOL1 * P(I)
XM2 = VOL2 * P(I)
XI1(I) = O(I) * 333.3 * X(I) * (F(I) * 2 + A(I) * 2)
XI2(I) = O(I) * 333.3 * X(I) * (C(I) * 2 + F(I) * 2)
IXCG(I) = (XI1(I) + XI2(I))
XI3(I) = O(I) * 333.3 * X(I) * (LGTH * 2 + A(I) * 2)
XI4(I) = O(I) * 333.3 * X(I) * (C(I) * 2 + F(I) * 2)
IYCG(I) = (XI3(I) + XI4(I))
XI5(I) = O(I) * 333.3 * X(I) * (LGTH * 2 + F(I) * 2)
XI6(I) = O(I) * 333.3 * X(I) * (C(I) * 2 + F(I) * 2)
IZZCG(I) = (XI5(I) + XI6(I))
GO TO 40

ORIGINAL PAGE IS OF POOR QUALITY
PARALLELEPIPED (THIN WALL) *SHAPE 10*

36 XL(I) = LGTH/2.
   *(I) = RH0(I).
   TEMP1 = (LGTH*B(I) + A(I)).
   TEMP2 = (LGTH*B(I) + B(I)*A(I) + LGTH*A(I)).
   IXCG(I) = 0.033333*RH0(I)*(B(I)**2 + A(I)**2)*(RH0(I)/6.0)*(TEMP1
   *(B(I) + A(I))/TEMP2).
   IYCG(I) = 0.033333*RH0(I)*(LGTH**2*A(I)**2)*(RH0(I)/6.0)*(TEMP1
   *(LGT + A(I))/TEMP2).
   IZCG(I) = 0.033333*RH0(I)*(LGTH**2 + B(I)**2)*(RH0(I)/6.0)*(TEMP1
   *(LGT + B(I))/TEMP2).
   GO TO 60.

SHEET TRAPEZOIDAL PANEL (THICK WALL) *SHAPE 11* (REF->R, HULL)

11 *(I) = A(I) + RH0(I) *(LGTH*(H(I) + F(I)))/2.0.
   IF (RH0(I) + ST0.0, 0) = I = RH0(I).
   RH0(I) = *(I) / (((B(I) + F(I))/2.0) * LGTH*A(I)).
   XL(I) = LGTH*(B(I) + 2.0*F(I)) / (3.0*(B(I) + F(I))).
   XT = F(I) *(LGTH*(B(I) + F(I))).
   THTAN = C(I)/LGTH.
   FETAN = (F(I)/6.0 + C(I)/LGTH).
   AFTAN = (C(I) - F(I) + B(I))/2.0)/LGTH.
   THTRE = LGTH*XT.
   TCG = (FETAN*AFTAN)/2.0)*(TTHRE*XL(I)).
   XXPAA = AFTAN.
   IYCG(I) = *(I) *(XPAA**2) + ((I) + 3.0*(TTHRE**2 - X**2))/2.0 + XPAV*(A(I) +
   2*(TTHRE**4 - X**4))/4.0) + I* (XFPAA**2)* (TTHRE**4 - X**4))/4.0 + XPAV*(A(I) +
   (TTHRE**4 - X**4))/4.0 + XPAV*(A(I) +
   I* (XFPAA**2)* (TTHRE**4 - X**4))/4.0 + XPAV*(A(I) +
   1.0 + FETAN + AFTAN + 2.0*(TTHRE**4 - X**4))/16.0).
   IYCG(I) = RH0(I) *(XPAV**2)* (A(I) + (TTHRE**4 - X**4))/6.0 + XPAV*(A(I)**2)
   1.0 + (TTHRE**4 - X**4))/16.0).
   IZCG(I) = A(I) + RH0(I) *(XPAV**3)* (TTHRE**4 - X**4)/4.0 + XPAV*(A(I) + (TTHRE**4 -
   3.0) + 2.0*(TTHRE**4 - X**4))/2.0 + (2.0*(TTHRE**4 - X**4))/16.0 + XPAV*(A(I) +
   (TTHRE**4 - X**4))/16.0).
   IF (APS(T(I))< 0.01) GO TO 60.
   APX = C(I) + 2.0 + AS(THTAN*XL(I)).
   N = APXCG + 0.01*(X - I)*Y(I)**2 + (Y(I) - Y(I)**2 + (Z(I) - Z(I)**2)).
   CGK = + APXCG.
   PRX = TTHRE*LGTH.
   SY = + PRX/THR3.
   PRTY = 5.0/F(I)**2 + 1.0 + (LGTH*PRX + 3.0 + PRX**2)/2.0).
   IYCG(I) = 0.033333*RH0(I)*(LGTH**2*A(I)**2)*(RH0(I)/6.0)*(TEMP1
   *(LGT + A(I))/TEMP2).
   GO TO 60.

42
```
IXYCG(I) = ABS(IYYCG(I)) * SIN(PAXRC) * COS(PAXRC) * 2 + IYYCG(I) * COS(PAXRC) * 2

IYBGC(I) = IXXCG(I) * SIN(PAXRC) * COS(PAXRC) * 2 + IYYCG(I) * COS(PAXRC) * 2

SXHFTI = ABS(TANG(PAXRC)) * N(I)

LTH = (LTH - 2.5 * N(I)) / 2.0

VOL = (LTH - (R(I) + F(I)) + A(I)) * 2

IF (R(I) + F(I) + A(I)) = LTH, THEN LTH = (R(I) + F(I)) / 3

IYCG(I) = (LTH - 2.0 * R(I) + F(I)) / (3.0 + R(I) + F(I))

2*(R(I) + F(I)) * 1.5 + A(I) = 2/12 * (2.0 + (R(I) + F(I)) + R(I) + F(I)) / (R(I) + F(I))

XT = F(I) * LTH / (R(I) + F(I))

IF (R(I) + F(I) + A(I)) = LTH, THEN LTH = (R(I) + F(I)) / 3

TV = CT / LTH

EFTA = (F(I) / 2.0 + C(I)) / LTH

AFTA = (C(I) + (R(I) + F(I)) / 12.0) / LTH

IF (R(I) + F(I)) = LTH, THEN LTH = 0.0

LET = (2.0 + (R(I) + F(I))) / (LTH * XT) / (XT / LTH + XT / LTH)

IYBGC(I) = (XT / LTH) * 2 + Y(I) * LTH * 2 + Z(I) * LTH

GO TO 10
```

C

```
SAY WHAT TRAPEZOIDAL PANELS (THICK < ALL) SHAPE 12* (REF=Q, HULL)
```

43

ORIGINAL PAGE IS OF POOR QUALITY
CURVED THIN WALL PANEL *SHAPE 13*

10 \( n(1) = 2 \cdot \text{LTH} \cdot A(1) \cdot B(1) \cdot C(1) \cdot \text{RHO}(1) \)

\[ \text{IF} (\text{RHO}(1) > 0, 4) = (I) \cdot \text{RHO}(1) \]

\[ \text{RHO}(1) = n(1) / (2 \cdot \text{LTH} \cdot A(1) \cdot B(1) \cdot C(1)) \]

\[ L = 2 \cdot \text{C}(1) \cdot A(1) \cdot B(1) \cdot \text{LTH} \cdot \text{RHO}(1) \]

\[ L = \text{LTH} \cdot 2 \cdot 12 \]

\[ X(1) = \text{LTH} / 2 \]

\[ X(1) = \text{LTH} / 2 \]

\[ \text{Y}(1) = \text{LTH} / 2 \]

\[ \text{Z}(1) = \text{LTH} / 2 \]

\[ \text{RHO}(1) \cdot A(1) \cdot \text{C}(1) \cdot B(1) \cdot \text{LTH} \]

\[ \text{Y}(1) = (2 \cdot \text{C}(1)) \cdot (2 \cdot \text{SIN}(C(I)) \cdot \text{C}(1)) / (2 \cdot \text{C}(1)) \]

\[ \text{Z}(1) = (2 \cdot \text{C}(1)) \cdot \text{LTH} \]

\[ \text{X}(1) = (2 \cdot \text{C}(1)) \cdot \text{LTH} \]

\[ \text{GO TO 30} \]

BEGIN
dicos

60 IF (ABS(X(I)), NE, 0, OR, (ABS(Y(I)), NE, 0, OR, (ABS(Z(I)), NE, 0,)

\[ Y(1) = \text{X}(1) \cdot \text{Y}(1) \]

\[ Y(1) = \text{X}(1) \cdot \text{Y}(1) \]

\[ Z(1) = \text{Z}(1) \cdot \text{Z}(1) \]

\[ Z(1) = \text{Z}(1) \cdot \text{Z}(1) \]

\[ T(1) = X(1) \cdot X(1) \]

\[ T(1) = X(1) \cdot X(1) \]

\[ T(3) = Z(1) \cdot Z(1) \]

\[ T(3) = Z(1) \cdot Z(1) \]

\[ T = X(1) \cdot Y(1) \]

\[ T = X(1) \cdot Y(1) \]

\[ \text{Z} \times \text{T} = T \times T \times T \]

\[ \text{Z} \times T = T \times T \times T \]

\[ T = \text{X} \times \text{Y} \times 2 + \text{Y} \times \text{Z} \times 2 \]

\[ T = \text{X} \times \text{Y} \times 2 + \text{Y} \times \text{Z} \times 2 \]

\[ L = \text{X} \times \text{Y} \times 2 \]

\[ L = \text{X} \times \text{Y} \times 2 \]

\[ L = \text{X} \times \text{Y} \times 2 \]

\[ L = \text{X} \times \text{Y} \times 2 \]

\[ \text{END DICOS} \]
ROTATE COMPONENT MOMENT OF INERTIA TO SYSTEM COORDINATES

\[ IX \times CO(I) = (IXSCG(1) \times (LX) \times YCG(I) \times (LY) \times ZCG(1) \times (LZ)) / CON5 \]
\[ IY \times YCG(I) = (IXSCG(1) \times (MX) \times YCG(I) \times (MY) \times ZCG(1) \times (MZ)) / CON5 \]
\[ IZ \times ZCG(1) = (IXSCG(1) \times (NX) \times YCG(I) \times (NY) \times ZCG(1) \times (NZ)) / CON5 \]
\[ IY \times YCG(I) = ((IXSCG(1) \times (LX) \times (MY) \times (MY) \times ZCG(1) \times (MZ)) / CON5 \]

CALCULATE COMPONENT CENTER OF MASS COORDINATES AND WRITE OUT

\[ XCG(1) = YI(1) \times XL(1) \times LX \]
\[ YCG(1) = YI(1) \times YL(1) \times MX \]
\[ ZCG(1) = YI(1) \times XL(1) \times MX \]
\[ XCG(1) \times YCG(I) \times ZCG(1) \]

WRITE(300, I, I, Y, Y, Z, Z, Z)

CALCULATE SYSTEM WEIGHT, SECOND MOMENT AT ORIGIN AND C.G.

\[ I = 2 \times I(1) \]
\[ IXSCG(I) = (IXSCG(1) \times (LX) \times YCG(I) \times (LY) \times ZCG(1) \times (LZ)) / CON5 \]
\[ IYSCG(I) = (IXSCG(1) \times (MY) \times YCG(I) \times (MY) \times ZCG(1) \times (MZ)) / CON5 \]
\[ IZSCG(I) = (IXSCG(1) \times (NZ) \times YCG(I) \times (NY) \times ZCG(1) \times (NZ)) / CON5 \]
\[ VM = YCG(I) \times VM(I) \times YCG(I) \times VM(I) \times YCG(I) \times VM(I) \]

COMPUTE SYSTEM C.G. COORDINATES

\[ X = YCG(I) \times VM(I) \times VM(I) \times VM(I) \times VM(I) \times VM(I) \]
\[ VM = YCG(I) \times VM(I) \times VM(I) \times VM(I) \times VM(I) \times VM(I) \]
\[ Z = YCG(I) \times VM(I) \times VM(I) \times VM(I) \times VM(I) \times VM(I) \]

2 FORMAT(100, SYSTEM DATA INSERTED HERE (TYPES, INERTIAS, SLUGS FT SQ LBS, C.G., INS, SECOND MOMENT=SLUG FT SQUARED) */*)

WRITE(6, 100) X, XX, YY, ZZ, XBAR, YBAR, ZBAR

FORMAT(11, 3, 0, 1, 1, 1, 1, 1, 1, XX, YY, ZZ, XBAR, YBAR, ZBAR)

ORIGINAL PAGE IS OF POOR QUALITY
TRANSFER MASS PROPERTIES TO SYSTEM C.G., SUM AND WRITE OUT

D070 IM1, IMAX
DELX=XC(G(I))=XBAR
DELY=YC(G(I))=YBAR
DELZ=ZC(G(I))=ZBAR
IXX=IXX+XXC(G(I))=XBAR
IYY=IYY+YYC(G(I))=YBAR
IZZ=IZZ+ZZC(G(I))=ZBAR
IXY=IXY+IXYC(G(I))=XBAR
IZX=IZX+IZXC(G(I))=ZBAR
70 IYZ=IYZ+IYZC(G(I))=ZBAR
WRITE(b,200)IXX, IYY, IZZ
200 FORMAT(12X3H1x16X3HIYY15X3HIZZ15X3HX15X3H1Y15X3H1Z15X3H1Y15X3H1Z/6F18.5/)

COMPUTE INERTIAS (EIGENVALUES) ABOUT PRINCIPAL AXES AND EACH AXIS
DIRECTION COSINES (EIGENVECTORS) AND WRITE OUT

PRINT 340
340 FORMAT(1X,*INERTIAS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WITH
1H AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPAL AXE
1S TO THE X, Y, AND Z SYSTEM AXES IN THAT ORDER*/*//)
MAX=3

#5
955 AR=(1,1)=IMX
956 AR(2,1)=ARR(1,1)=IMX
957 AR(1,3)=ARR(1,3)=IXZ
958 AR(2,2)=IYY
959 AR(3,2)=ARR(3,2)=IYZ
960 AR(3,3)=IZZ
CALL SYV7L(MAX, AR, E, CR, IERR)
IF(IERR .NE. 0) GO TO 332
2=356 J=1,3
PRINT 357, J, E(J)
337 FORMAT(1X,*EIGENVALUES(*1)=*12.5//)
PRINT 339, J
339 FORMAT(1X,*EIGENVECTORS(*1)=//)
PRINT 33A, (ARR(I,J), I=1,3)
33A FORMAT(1X,3F14.6,5X)///
33B CONTINUE
GO TO 330
332 PRINT 333, IERR
333 FORMAT(1X,*ERROR = IERR = *15)
334 STOP
END
A method is presented for determining the complete mass properties of a rigid structure.

Computer technology is utilized to exercise the greatest savings possible in solving for and printing out this information including properties relative to the total structure's principal axes.